Feedback Vertex Set Problem

Feedback Vertex Set Problem (FVSP):
Given: Undirected graph $G = (V, E)$, weights $w : V \to \mathbb{R}_+$.
Task: Find $S \subseteq V$ of minimum weight s.t. $G - S$ is acyclic.

Note that S must "hit" every cycle in G.Let $\mathcal{C}$ (as sets of vertex sets) be the collection of all cycles in G.

IP-Formulation of (FVSP):

$$\begin{aligned} \text{OPT} = \quad \min \quad & \textstyle\sum_{i \in V}^{m} w_i x_i \\ \text{s.t.} \quad & \textstyle\sum_{i : i \in C} x_j \geq 1 \qquad \forall C \in \mathcal{C} \\ & x_i \in \{0, 1\} \qquad \forall i \in V \end{aligned} \tag{17.3}$$

Dual of linear relaxation:

$$\max_{y \geq 0} \{ \sum_{C \in \mathcal{C}} y_C \mid \sum_{C \in \mathcal{C} : i \in C} y_C \leq w_i, \ \forall i \in V \}$$

Choosing Variables to Increase

(Basic) Primal-Dual Algorithm for (FVSP):

1. Initialize: $y \equiv 0$; $S = \emptyset$;
2. WHILE $\exists$ cycle $C \in \mathcal{C}$ DO
3. Increase y_C until $\exists i \in V$ with $\sum_{C \in \mathcal{C} : i \in C} y_C = w_i$;
4. Set $S = S \cup \{i\}$;
5. Drop i from G;
6. Repeatedly remove vertices of degree one from G;
7. RETURN S;

Note: At most $n := |V|$ iterations.

Observation: α-approximation algorithm if $|S \cap C| \leq \alpha$ for each cycle C.

Proof: ... □

Lemma 17.9.
For any path/cycle P of degree-two vertices in G, the algorithm selects at most one vertex of P.

Proof: ... □

Crucial Lemma

Idea: Select cycle with small number of vertices of degree ≥ 3

Lemma 17.10.

In any graph G with no degree-one vertex, there is a cycle with at most $2\lceil \log_2 |V| \rceil$ vertices of degree ≥ 3. Moreover, such a cycle can be found in linear time.

Proof: … □

Primal-Dual Algorithm for (FVSP)

Primal-Dual Algorithm for (FVSP):

1. Initialize: $y \equiv 0$; $S = \emptyset$;
2. Repeatedly remove vertices of degree one from G;
3. WHILE $\exists$ cycle $C \in \mathcal{C}$ DO
4.   Find $C \in \mathcal{C}$ with at most $2\lceil \log_2 |V| \rceil$ vertices of degree ≥ 3;
5.   Increase y_C until $\exists i \in V$ with $\sum_{C \in \mathcal{C}: i \in C} y_C = w_i$;
6.   Set $S = S \cup \{i\}$;
7.   Drop i from G;
8.   Repeatedly remove vertices of degree one from G;
9. RETURN S;

Theorem 17.11.

The algorithm is a $(4\lceil \log_2 |V| \rceil)$-approximation algorithm for (FVSP).

Proof: … □

For this IP-formulation, no better performance guarantee can be obtained (unless $\mathcal{P} = \mathcal{NP}$). However, for a more sophisticated IP-formulation of (FVSP), there exists even a primal-dual 2-approximation algorithm.

Lattice Polyhedra

Recall: Lattice polyhedra provide a common framework for various efficiently solvable discrete optimization problems.

Definition 17.12.

Let E finite set, $\mathcal{F} \subseteq 2^E$ and $r : \mathcal{F} \to \mathbb{R}_+$. Then $\{x \in \mathbb{R}^E_+ \mid x(S) = \sum_{e \in S} x_e \geq r(S),\ \forall S \in \mathcal{F}\}$ is a lattice polyhedron if $(\mathcal{F}, \preceq, \wedge, \vee)$ forms a lattice s.t. for all $S, T, U \in \mathcal{F}$

1. $S \preceq T \preceq U$ implies $S \cap U \subseteq T$,
2. $(S \wedge T) \cup (S \vee T) \subseteq S \cup T$,
3. $r(S) + r(T) \leq r(S \wedge T) + r(S \vee T)$.

Examples: ...

Primal-Dual Algorithm for Lattice Polyhedra

Given $c; E \to \mathbb{R}_+$ consider the (dual) linear program:

$$\max_{y \geq 0}\{\sum_{S \in \mathcal{F}} r(S) y(M) \mid \sum_{S : e \in S} y_S \leq c_e,\ \forall e \in E\}.$$

Dual algorithm:

1. Initialize: $y^* \equiv 0$;
2. WHILE $\mathcal{F} \neq \emptyset$ DO
3.   Select the $\preceq$-maximal set $M \in \mathcal{F}$ (unique!);
4.   Increase $y^*(M)$ until some $e^* \in M$ becomes tight;
5.   Set $\mathcal{F} = \{S \in \mathcal{F} \mid e^* \notin \mathcal{F}\}$;
6. RETURN y^*;

Theorem 17.13.

If $(\mathcal{F}, r)$ is a lattice polyhedron and $r(S) \leq r(T)$ whenever $S \preceq T$, then y^* is optimal.